



## Deploy de la aplicación



### Transcripción

[00:00] Hola a todos y todas, bienvenidos y bienvenidas una vez más a este curso de Servlet. En el video pasado tuvimos un poco más sobre teoría y en este caso vamos a ir un poco más a la práctica, con lo que es un deploy. ¿Qué es un deploy?

[00:14] Significa enviar todo nuestro código, toda nuestra aplicación a otra máquina, una máquina en producción, en la cual esa máquina en producción es un lugar, una máquina en la que por ejemplo nuestro cliente va a tener su servidor, ya sea en la Internet o en la nube, por ejemplo, en la nube de Amazon, Google, Microsoft.

[00:33] Entonces es básicamente otra computadora normalmente. Entonces en nuestro caso nosotros vamos a simular eso, creando una nueva instancia de Tomcat, ya que por ejemplo hacer un ejemplo enviando nuestro servidor, nuestro código a la nube de Amazon, no sería muy conveniente, ya que en este caso los servidores de Amazon son pagos.

[00:59] Entonces vamos a solo simular ese deploy. Bien para eso, ¿qué es lo que vamos a hacer? Vamos a ir a nuestro archivo que nosotros ya teníamos de nuestro apache-tomcat que lo bajaron desde el comienzo del curso, vamos a hacer clic derecho, copiar y lo voy a copiar en mi disco D. Pegar.

[01:19] Acá yo lo voy a extraer, clic derecho, extraer, bien, y voy a eliminar el punto zip y me quedo acá con nuestra carpeta de apache-tomcat con una instalación limpia. Bien, entonces con esto por ejemplo estamos

simulando eso, un servidor de Tomcat en la nube y por ejemplo como curiosidad, Alura funciona como un Tomcat en la nube de Amazon.

[01:46] Entonces, bueno, cómo hacemos para pasar todos nuestros archivos a nuestro Tomcat, nuestra nueva versión de Tomcat, nuestra nueva instancia de Tomcat. Sería muy, muy engorroso, muy difícil tener que copiar todos los archivos y pegarlos, y no es así como funciona normalmente un deploy.

[02:07] Normalmente en un deploy lo que nosotros hacemos es exportar un archivo, así como por ejemplo dentro de nuestra librería nosotros tenemos esos archivos jar de nuestros JSTL, mMe gustaría tener algo parecido, solo que para nuestra aplicación web. Entonces para eso lo que se inventó dentro de la tecnología toda esta de webapps es archivos WAR.

[02:34] Esos archivos WAR es muy parecido a los archivos JAR o sea Java archive, solo que en este caso es web archive. Es otro padrón, pero básicamente tienen la misma estructura. Nosotros si por ejemplo cambiamos su extensión a .zip, nosotros podríamos ver todos los archivos que tienen entre medio.

[02:57] Otra cosa que me gustaría hablar sobre lo que es el deploy es que yo podría tener, por ejemplo, muchas instancias de Tomcat diferentes. En vez de tener solo nuestro Tomcat de desarrollo, que es lo que estuvimos haciendo a lo largo de todo el curso, y un Tomcat de producción, que es lo que vamos a hacer ahora.

[03:15] Podría tener, por ejemplo un Tomcat para test, para que haya personas QAs, por ejemplo, personas que hacen exámenes de calidad de nuestra aplicación y por ejemplo podría tener otra instancia para homologación, por ejemplo que sea una etapa antes de producción para no enviar código directamente a producción.

[03:36] Entonces podremos tener muchas instancias. En este caso solo vamos a crear una más de producción. Entonces vamos a hacer eso, vamos a crear

nuestro archivo WAR. Vamos a nuestro Eclipse. En nuestro gerenciador, nuestro proyecto gerenciador clic derecho y acá vamos a hacer un export.

[03:58] Dentro de ese export tenemos acá directamente un WAR file, pero si ustedes no ven esa opción, pueden ir a export y acá escribir WAR. Perfecto. tenemos nuestro WAR file, entonces vamos a hacer clic en next, el proyecto que queremos exportar es gerenciador, tenemos que decirle dónde queremos exportarlo. En este caso yo voy a colocar que esté en el disco D, en el mismo lugar que nuestro apache-tomcat, solo que bueno, no dentro de apache-tomcat sino dentro del mismo directorio.

[04:31] Entonces vamos a hacer guardar y todo esto lo dejamos como padrón. Finish, ahora en teoría dentro de mi Disco D, yo tendría un archivo WAR. Acá tengo ese archivo, ese gerenciador.war. Si por ejemplo les dije que ese WAR es muy parecido con JAR y nosotros a todos esos archivos los podemos abrir con un winrar winzip, entonces, por ejemplo, podríamos renombrar acá zip y podríamos ver qué es lo que tiene adentro.

[05:01] Doble clic y vean lo siguiente, nosotros tenemos los mismos archivos que tenemos dentro de nuestro webapps. Dentro de webapp nosotros tenemos todos esos archivos también. Vean que tenemos META o WEB-INF y tenemos todos los JSP, HTML.

[05:18] Y dentro de WEB-INF nosotros tenemos lo mismo que teníamos acá en WEB-INF nuestra librería y los web.xml, esas librerías con nuestro JAR, web.xml, pero también tenemos una carpeta llamada clases. ¿Qué es lo que tenemos dentro de esas clases? Tenemos todo lo que son nuestros Servlets, todo lo que tenemos acá.

[05:40] Entonces están dentro de clases que ya están compilados. Entonces, si vamos a esa carpeta, vean que tenemos archivos .class y no .java, que era lo que teníamos acá. Entonces ya están compilados. Perfecto, con esto yo voy a

volver acá a colocar archivo WAR, cambio la extensión y voy a moverlo ¿a dónde? "Ctrl + X" para cortarlo y luego enviar en Tomcat.

[06:11] Y una cosa que me gustaría decirles: no existe solamente un tipo de servidor, existe así como esta Tomcat existen otros servidores, y muchos de ellos ya aceptan los archivos .WAR, entonces es una extensión que se puede utilizar para subir a otros servidores, entonces simplemente hay que ver si ese servidor acepta ese tipo de archivos WAR. ¿Qué hacemos acá?

[06:40] Nosotros tenemos nuestro archivo WAR y nosotros lo tenemos que enviar a ese webapps. En otros servidores, normalmente la carpeta en la que tenemos que colocar ese WAR se llama deploy, pero en este caso Tomcat decidió llamarla webapps. Simplemente hay que aceptarlo.

[06:57] Entonces, dentro de ese webapps ya tendremos nuestro archivo .WAR. ¿Ahora qué es lo que tenemos que hacer? Nosotros tenemos que ir a nuestra consola de comandos y llegar hasta acá no, tenemos que ir a hasta nuestro apache-tomcat. ¿Cómo hacemos eso?

[07:13] En Windows tenemos un atajo, que es con "Shift + clic derecho", en la parte vacía, abrir la ventana de powershell aquí y ya automáticamente estamos acá. Si están en Linux, por ejemplo, ustedes pueden ir haciendo el CD y llegar hasta la carpeta que necesitamos dentro de su apache-tomcat.

[07:33] Dentro de este apache-tomcat y en la mayoría de los programas tenemos una carpeta llamada bin. Ahí es donde nosotros tenemos nuestro ejecutable de, en este caso Tomcat. Entonces vamos a hacer un ciudad bin y acá en Windows vamos a hacer un dir o en Linux hace un ls y vemos los archivos que tenemos.

[07:55] Nosotros tenemos que ejecutar este. En Windows es startup.bat y si están en Linux o en Mac usan startup.sh. Yo voy a usar startup.ar, startup.bat, o sea, \startup.bat. Vean lo que ocurre. En mi caso, yo no tengo Java instalado, y

ahora pueden preguntarse ¿cómo que no tenemos Java instalado, si Eclipse cuando nosotros lo instalamos, instalamos Java también?

[08:24] La cuestión es que esa instalación es propia de Eclipse y ese Java que nosotros instalamos fue a parar a una carpeta propia de Eclipse. Entonces acá ustedes puede ser que tengan esa misma situación, que no tengan Java instalado. O puede ser que tengan sí Java instalado. La cuestión es que necesitan una versión en particular.

[08:45] Necesitan tener en particular la siguiente versión. Vamos a ir a nuestro gerenciador dentro de nuestro Eclipse. Gerenciador, clic derecho, propiedades Y dentro de Java Compiler acá ven cuál es la versión que ustedes tienen. En este caso yo tengo la versión Java SE 17. Entonces lo que voy a hacer es buscar en Internet esa versión para instalarla en Windows.

[09:08] En este caso yo ya había hecho la búsqueda. Simplemente colocando Java SE y el nombre de la versión aparece dentro del primer resultado la página de Oracle y la opción para bajar ese Java SE. Entonces voy a entrar acá y como yo estoy en Windows voy a usar la instalación x64.

[09:28] Entonces bajo esa versión y voy a proceder a instalarlo. La instalación es típica, es normal, esto es todo siguiente, siguiente, siguiente, no van a tener problema. En mi caso vean una cosa, se va a instalar en C:\Program Files\Java, eso es un dato importante que lo van a necesitar después.

[09:51] Luego de la instalación, hacen un close y ahora ustedes están pensando perfecto, puedo usar \startup.bat. Pero no. ¿Qué es lo que ocurre? No tenemos nuestras variables de ambiente configuradas. ¿Qué es esa variable de ambiente? Existen tanto en Windows como en Linux. Entonces ustedes tienen que configurarlas.

[10:13] Una vez que tienen instalado el Java, tienen que decirle a Windows, al sistema operativo o a Linux, pueden utilizar, o sea, ustedes le tienen que decir



a la máquina que el sistema operativo puede utilizar el Java que está en la siguiente carpeta. Entonces para eso, ¿cómo hacemos?

[10:33] Apretamos la tecla Windows en este caso, comenzamos a escribir variables, vean que existe la opción de editar las variables del entorno del sistema. Hacemos click ahí. Y acá adentro vamos a variables del entorno y acá abajo, en esta variable del sistema, nosotros tenemos que crear una nueva variable. ¿Cómo hacemos eso?

[10:56] Nueva y acá colocamos todo en mayúsculas: JAVA\_HOME y acá le tenemos que decir en dónde es que está nuestro Java, entonces vamos a hacer clic en examinar directorio. Voy a irme a mi equipo, C, recuerdan que les dije que tienen que fijarse bien dónde va a instalarse. Bueno en program files. Y acá yo tengo mi Java y tengo mi Java 17. Tienen que colocar esa carpeta y no la carpeta bin. Esa carpeta.

[11:28] Aceptar, aceptar. Aceptamos también, aceptamos también. Cierran la consola de comandos porque si no, no va a actualizar las variables de ambiente. Vamos de nuevo a nuestra carpeta de Tomcat, clic derecho. Abrir la ventana de PowerShell aquí, y ahora vamos de nuevo a nuestro bin, cd bin, y recuerden que tenemos que ejecutar ese startup.bat.

[11:57] Entonces vamos a hacer startup start, apretamos la tecla tab para autocompletar eso creo que no lo dije antes, disculpen, y hacemos clic en enter. Bien, vean lo siguiente. Ya se ejecutó, o sea, está ejecutándose y nos abrió una nueva consola de comandos, un cmd de dentro de Windows, Linux va a ser una terminal.

[12:19] Bien, esto significa que ya está funcionando nuestro servidor. Si no aparece ningún error dentro de acá de la consola, ustedes siempre miren, a ver si no existe algo que diga error o alguna cosa así, si les ocurre algún error, pueden escribir dentro de nuestros foro y ahora vamos a testar. Veán que

tenemos la versión de Java jdk-17 instalada, que es la misma que teníamos acá en nuestro gerenciador, que es la 17.

[12:47] Es importante eso, ya que si tienen versiones diferentes de Java puede ser que les aparezca algún problema dentro de la consola de comandos al abrir algunas de las páginas que vamos a abrir de nuestro gerenciador. Entonces si ocurre eso, simplemente tienen que cambiar, bajar la versión de Java que corresponde a su proyecto y cambiar las variables de ambiente, que son estas variables de acá.

[13:16] Solo cambian el camino hacia y apuntan hacia la versión de Java que utilizamos en nuestro proyecto y listo. Con eso ya debería funcionar esa parte. Perfecto, está funcionando esto y vamos a probar, vamos a probar nuestro proyecto. Vean que nuestro Eclipse, en nuestro Eclipse de desarrollo está parado nuestro servidor, entonces no estoy usando nuestro servidor.

[13:40] Vamos a ir acá, abro una nueva pestaña localhost, voy a primero, ir al localhost:8080, vean lo siguiente, nos está contestando un Apache Tomcat, una página bien bonita, entonces significa que Tomcat por lo menos está funcionando. Vamos a ver qué es lo que pasa con nuestro gerenciador, a ver si está funcionando.

[14:01] Gerenciador está funcionando bien, perfecto, buenísimo. Ya estamos, ya hicimos un deploy de nuestra aplicación a otra instancia de Tomcat que, como les había dicho, podría ser en la máquina de un cliente o en la nube, por ejemplo, en cualquier lugar. Vamos a probar, por ejemplo, nuestra lista de empresas, a ver si está funcionando.

[14:26] Está funcionando. Vamos a hacer modificar, voy a colocar Aluraaaaaa con muchas A, cambiar perfecto. Funciona nuestro modificar. Eliminar también funcionó. Entonces está funcionando todo esto, esto está bastante, bastante bien. Recuerden si tienen algún problema escriban en el foro.

[14:44] Esta parte puede ser un poco complicada porque depende mucho de nuestra computadora en la que nosotros estamos, entonces es importante eso. Perfecto. Nosotros hicimos ahora un deploy manual, todo el proceso lo hicimos manualmente, pero normalmente en un proyecto con alguna empresa nosotros normalmente ya tenemos scripts que hacen todo eso automáticamente.

[15:07] Todo eso forma parte del mundo de DevOps. Ahora si nosotros, por ejemplo, quisiéramos modificar nuestra aplicación, nosotros tendríamos que ir a nuestro Eclipse y realmente modificar nuestro Servlet o lo que queramos y hacer de nuevo ese proceso de deploy.

[15:24] Clic derecho y generar un nuevo WAR y enviar nuestro WAR, nuestro Tomcat, nuestra instancia Tomcat, etcétera. Ahora quedó una cosa que no me gustó. No me está gustando que tengamos que poner ese 8080 ahí, es nuestra puerta, normalmente uno cuando va a la web no coloca 8080, nosotros solo colocamos localhost/gerenciador.

[15:50] Y acá no nos está contestando, si nosotros hacemos eso. Entonces vamos a hacer ese cambio, vamos a ir a nuestro Tomcat, a nuestro Tomcat de producción. Dentro de ese Tomcat nosotros tenemos una carpeta llamada conf y dentro de esa carpeta tenemos nuestro, entonces vamos a hacer un clic derecho, voy a editarlo con mi notepad++, pueden editarlo con cualquier editor de código.

[16:14] Y acá, para no tener que buscarlo manualmente, "Ctrl + F" y vamos a colocar 8080 y vamos a ver dónde es que está. Acá tenemos nuestro 8080, donde definimos cuál es la puerta. Entonces en vez de 8080 vamos a colocar 80 solamente, voy a hacer un "Ctrl + S" para guardar el archivo, y con esto ya deberíamos poder simplemente colocar localhost/gerenciador.

[16:40] Tengo que cerrar nuestro Tomcat. Vean que acá aparecieron las variables, lo que nosotros sacábamos antes por la consola, con el System.out,



aparecen también acá en esta consola. Vamos a cerrar nuestro Tomcat y voy a abrirlo de nuevo, ejecutando el mismo startup.bat.

[17:02] Con esto carga de nuevo las configuraciones bien, y ahora sí podría, en teoría ahí está, colocar localhost/gerenciador. ¿Por qué puede hacer esto? Porque todas las páginas normalmente están en el puerto 80. Entonces, cuando no colocamos ningún puerto acá, automáticamente el navegador entiende que tiene que ir al puerto 80.